

Simulador de Hundir la flota

Información de contacto del profesor:


Jordi Bayo Singla

jordi@iceverg.com


Linkedin => <https://www.linkedin.com/in/jordi-bayo-singla-a945581b/>

Discord => jbs#4215

Hundir la flota




www.lalibretapiruleta.com



Mis barcos

J									
I									
H									
G									
F									
E									
D									
C									
B									
A									
	1	2	3	4	5	6	7	8	9 10

Barcos:



Sus barcos

J									
I									
H									
G									
F									
E									
D									
C									
B									
A									
	1	2	3	4	5	6	7	8	9 10

Barcos:

Para consolidar nuestros conocimientos de JavaScript, tendréis que desarrollar un **simulador del popular juego de mesa Hundir la flota**.

Juego de Hundir la flota

Popularmente también conocido como Battleship, batalla naval, hundir los barcos... El juego trata de 2 jugadores que disponen de dos tableros, el propio y el del adversario. Gana el jugador que hunde primero los barcos de su adversario. Para más información tenéis [https://es.wikipedia.org/wiki/Batalla_naval_\(juego\)](https://es.wikipedia.org/wiki/Batalla_naval_(juego)) que os lo explica de maravilla.

Simulador del juego (parte obligatoria)

El simulador deberá representar la partida de dos jugadores. Las características del juego serán las siguientes:

- 2 Jugadores
- 2 tableros por jugador (el propio y el del adversario)
- Los tableros tendrán 10 filas y 10 columnas
- Las naves a colocar son las siguientes:
 - 1 Portaaviones (5 casillas)
 - 1 Buque (4 casillas)
 - 2 Submarinos (3 casillas)
 - 3 Cruceros (2 casillas)
 - 3 lanchas (1 casilla)
- La cantidad de disparos de cada jugador permitidos son 10 filas x 10 columnas = 100 disparos

El desarrollo del juego

El simulador va a tener 2 fases bien diferenciadas:

1. Colocación de las naves en el tablero propio
2. Desarrollo del juego por turnos
 - a. En el momento que un jugador gane, termina el juego y se muestra por pantalla qué jugador ha ganado

Para que tenga un poco de sentido, deberemos implementar el código siguiendo estas pautas de funcionamiento.

Fase de colocación de las naves en el tablero

Está permitido:

- Se pueden poner barcos pegados entre sí, por ejemplo, tendríamos 1 submarino vertical (A1, B1, C1) y 1 lancha pegada al lado (B2).

(index)	0	1	2	3
A
B	.	x	x	.
C
D

No está permitido:

- **La posición de los barcos en diferentes ejecuciones del simulador no puede ser la misma.** Tiene que haber un poco de aleatoriedad en la colocación de los barcos. Por lo tanto, no puede ser que los barcos tengan siempre la misma distribución en la parrilla.

- **No se pueden solapar los barcos** (es decir, una plaza ya ocupada por un barco no puede ser ocupada por otro barco). Ya sé que los submarinos van por debajo del agua, pero no cuela, no pueden solaparse.
- **Las naves no pueden estar colocadas en diagonal**, únicamente en horizontal o en vertical.
- **Las naves que ocupan más de 1 casilla, no pueden dividirse**. Por ejemplo, no se puede poner un portaaviones así (hay un salto de línea entre B y C)

(index)	0	1	2	3	4	5	6	7	8	9
A										
B										
C										
D										
E										
F										
G										
H										
I										

Ideas:

- Intentad colocar primero las piezas más grandes primero, como si se tratara de llenar un bote de piedras; empezaríamos por las mayores hasta llegar a la arenilla.
- Si os encalláis con el paso de la colocación, preguntad por discord, recordad que tenéis un profe y compis para sacarlo adelante.

Fase de desarrollo del juego

El desarrollo es bien sencillo: empieza disparando un jugador indicando a qué coordenada dispara (del tablero del contrincante), si no le da a una nave contrincante será “Agua” y pasará el turno de disparo al otro jugador; si en cambio da a una nave del oponente, será entonces “Tocado”. Si además no quedan más casillas de esa nave “tocadas”, adicionalmente deberá mostrarse “Hundido”. En el caso de que se toque alguna nave contrincante, el jugador que ha disparado podrá seguir disparando hasta que falle (“Agua”). En ambos casos, por cada disparo deberá marcarse en el tablero del contrincante la coordenada seleccionada para el disparo.

El juego terminará cuando no haya más disparos o bien cuando uno de los dos jugadores haya hundido toda la flota contrincante (en vuestro caso es superfluo).

Para ordenar un poco más las ideas, tenéis esta simplificación en pseudocódigo:

Mientras no haya ganador:

 Turno jugador:

 Mientras haya barcos del Jugador contrario:

 Jugador 1 (J1) → selección de coordenada del tablero contrario

 Mirar qué hay en el tablero propio del Jugador 2 (J2)

 si agua

 marcar agua en tablero contrario (J1)

 marcar agua en tablero propio (J2)

 basta jugador 1, turno del jugador 2

 si tocado

 marcar tocado en tablero contrario (J1)

 marcar tocado en tablero propio (J2)

 si hundido

 contar naves hundidas

 si todas naves hundidas

 basta jugador 1

Está permitido

- La selección de casillas oponentes de forma aleatoria, siempre y cuando respete el siguiente punto (no permitido)

No está permitido

- **Seleccionar casillas de barcos directamente del oponente.** Esto sería hacer trampas, mirar el tablero del oponente para saber dónde están sus naves...
- **Seleccionar casillas que ya han sido seleccionadas previamente.** No podéis disparar ahí donde habéis disparado.
- **No puede haber transiciones extrañas**, como que un barco reflote. Las transiciones que puede tener una casilla son
 - Vacío → tocado
 - Vacío → agua
 - Barco → tocado
 - Tocado → hundido

Ideas:

- Intentad escribir vosotros solos el pseudocódigo, transformarlo en comentarios y de ahí a la implementación.
- Si no acabáis de tener vuestra propia versión de pseudocódigo, podéis usar el que se os propone en forma de comentarios para luego ir implementando la solución del juego

Inteligencia (parte opcional)

Para poder dotar de algo más de inteligencia al juego, cuando una casilla es “tocada” y “no hundida” la selección posterior de la casilla podría ser alguna adyacente (vertical u horizontal) ya que los barcos de más de una casilla tienen esa distribución. Podéis usar algo de “memoria” para ir a derribar cuanto antes esa nave que ya tenéis identificada como tocada cuando vuelva a ser el turno del jugador.

Para sumarle algo de complejidad, en este caso deberá tenerse en cuenta el número de disparos posibles por jugador, de forma que se pueda reducir. Esto genera la posibilidad de que haya empate. Si hay empate, gana entonces el jugador que tenga más casillas de barcos sin tocar. En caso de empate otra vez, entonces no gana nadie.

Directrices generales

Está permitido

- Que los tableros no tengan letras, puede ser 0..9 tanto para filas como para columnas
- Podéis usar cualquier representación para los tableros, usando `console.log` o `console.table`. Si usáis `console.table` indicad por favor cuál es el tablero propio y cuál es el tablero contrincante para cada jugador.
- No hace falta mostrar por pantalla los barcos de cada jugador con sus estatus
- Usar funciones o código visto en clase.

Si tenéis dudas, preguntadlas.

EMPEZAD CON LA PARTE OBLIGATORIA!!! En cuanto tengáis la parte obligatoria, cercioraos de que pasáis los criterios de aceptación para que sea APTA. Una vez lo tenéis claro, podéis probar con la parte opcional, pero tened una copia de la obligatoria por si os liáis con la opcional.

Requisitos del programa mínimo

- El programa debe tener un `index.js` que será lo que ejecutaré
- El programa empiece con una frase indicando que empieza el juego (el idioma no importa)
- Las casillas deben tener los siguientes posibles representaciones (podéis usar cualquier representación que queráis siempre y cuando sean diferentes entre sí):
 - vacío
 - agua
 - barco (puede ser una única para todos los barcos o diferente para cada tipo de barco)
 - tocado
- El programa deberá mostrar la colocación de las naves en el tablero propio del Jugador 1 y a continuación mostrar las naves del tablero propio del Jugador 2
- El programa deberá mostrar una frase indicando que empiezan las rondas
- Para cada turno, deberá mostrarse claramente el turno del jugador X
- Para cada disparo, se deberá mostrar la casilla seleccionada, los disparos que faltan del jugador, y los tableros del jugador que está disparando (el propio y el ajeno) con la casilla ya marcada, sea agua o tocado.
- Cuando termine la partida, deberá mostrarse por pantalla qué jugador ha ganado.

- Deberá mostrarse también por pantalla el tablero propio del jugador 1 y el tablero propio del jugador 2, para poder visualizar los barcos que han quedado sin hundir

Requisitos el programa opcional

- Las celdas adyacentes deben ser “dentro” del tablero, no pueden haber saltos de línea (ver ejemplo de división de naves)
- Se deberá mostrar qué jugador gana por KO
- En caso de haber ganador que no es por KO, mostrar el recuento de casillas de barcos a flote de cada jugador.
- En caso de empate definitivo mostrar un texto de empate.

Calificación

La práctica será APTA siempre y cuando se cumplan TODOS los requisitos del programa mínimo respetando las prohibiciones de la fase de colocación y de la fase de desarrollo.

```
if (TodosRequisitosOK && TodasRestriccionesOK) {  
  
    practica = 'APTA'  
  
} else {  
  
    practica = 'NO APTA'  
  
}
```

Si la entregáis con la parte opcional, no la tendré en cuenta para la calificación del APTO, NO APTO, pero obviamente recibiréis feedback al respecto (recibiréis feedback de lo que entregáis y quizás de lo que no entregáis también)

[illegible]

(index)	A	B	C	D	E	F	G	H	I	J
0										
1										
2										
3										
4										
5										
6										
7										
8										
9										

Round 0 for B

=====

Shoot #1 pointing to 0C:

Own board:

(index)	A	B	C	D	E	F	G	H	I	J
0										
1										
2										
3										
4										
5										
6										
7										
8										
9										

Enemy board:

(index)	A	B	C	D	E	F	G	H	I	J
0										
1										
2										
3										
4										
5										
6										
7										
8										
9										

Round 1 for A

=====

Shoot #2 pointing to 3J:

Own board:


(index)	A	B	C	D	E	F	G	H	I	J
0										
1										
2										
3										
4										
5										
6										
7										
8										
9										

Enemy board:
































(index)	A	B	C	D	E	F	G	H	I	J
0
1
2
3
4
5
6
7
8
9

Round 1 for B



=====

Shoot #2 pointing to 9H: 

Own board:


(index)	A	B	C	D	E	F	G	H	I	J
0
1
2			
3	.									
4	.			.			.			.
5	.			.			.	.		.
6		
7			.	.	.	
8		
9	

Enemy board:






































(index)	A	B	C	D	E	F	G	H	I	J
0	.	.	
1
2
3
4
5
6
7
8
9		.	.

Round 2 for A

=====

Shoot #3 pointing to 5G: 

Own board:

(index)	A	B	C	D	E	F	G	H	I	J
0	.	.	
1		.	.							.
2		.	.		.					.
3	.	.	.		.				.	.
4	.	.	.		.					.
5	.		.	.	.					.
6					.	.
7	.	.	.			.			.	.
8
9		.	.

Enemy board:

(index)	A	B	C	D	E	F	G	H	I	J
0
1
2
3	💧
4
5	💧	.	.	.
6
7	💧
8
9

Round 2 for B

=====

Shoot #3 pointing to 1D: 💧

Own board:

(index)	A	B	C	D	E	F	G	H	I	J
0
1
2	🚢	🚢	🚢	🚢	🚢	🚢	🚢	🚢	🚢	💧
3	.	🚢	🚢	🚢	🚢	🚢	🚢	🚢	🚢	.
4	.	🚢	🚢	.	🚢	🚢	.	🚢	🚢	.
5	.	🚢	🚢	.	🚢	🚢	💧	.	🚢	.
6	🚢	🚢
7	🚢	🚢	.	.	.	💧
8	🚢	🚢
9	🚢	🚢

Enemy board:

(index)	A	B	C	D	E	F	G	H	I	J
0	.	.	💧
1	.	.	.	💧
2
3
4
5
6
7
8	💧	.	.
9

. . .
. . .
. . .

Round 75 for A

=====



































































































Shoot #99 pointing to 2C: 🌟 and drawn!

Own board:

(index)	A	B	C	D	E	F	G	H	I	J
0	💧	💧	💧	💧	💧	💧	💧	💧	💧	.
1	🌟	💧	💧	💧	🌟	🌟	🌟	🌟	🌟	💧
2	🌟	💧	💧	🌟	💧	💧	💧	💧	💧	💧
3	💧	💧	💧	🌟	💧	💧	🌟	🌟	🌟	💧
4	💧	💧	💧	🌟	💧	💧	🌟	🌟	🌟	💧
5	💧	💧	💧	💧	💧	🌟	🌟	🌟	🌟	💧
6	💧	🌟	💧	💧	💧	🌟	🌟	🌟	💧	💧
7	💧	💧	💧	🌟	🚢	💧	💧	💧	💧	💧

8										
9										

Enemy board:

(index)	A	B	C	D	E	F	G	H	I	J
0										
1										
2										
3										
4										
5										
6										
7										
8										
9										





































































































And the winner is.....

```
=====
===== Player A =====
=====
```

And the final boards are


























































































Player A

Own board:

(index)	A	B	C	D	E	F	G	H	I	J
0										
1										
2										
3										
4										
5										
6										
7										
8										
9										

Player B

Own board:

(index)	A	B	C	D	E	F	G	H	I	J
0										
1										
2										
3										
4										
5										
6										
7										
8										
9	